A Framework for Combining Ontology and Schema Matchers with Dempster-Shafer

Paolo Besana

School of Informatics
University of Edinburgh

Abstract Ontologies, at least in the form of taxonomies, have proved rather successful, and are employed in many fields, as far apart as biology and finance. Reaching an agreement over a single ontology has proved difficult, and to obtain actual interoperability it is necessary to map the different ontologies. Mapping one entity between a source ontology and one in a target ontology means to compare the first entity with all the entities in the second ontology: matchers analyse different aspects of the entities to identify the similarities. A single matcher can analyse only some aspects, and often has to rely on uncertain information. Therefore combining the outcomes of different matchers can yield better results.
In this paper I present a framework that uses Dempster-Shafer as a model for interpreting and combining results computed by the matchers.

1 Introduction

Ontologies have proved to be a powerful tool, and they have become common. For example, ontologies in the form of taxonomies are used by Google and Yahoo to categorise websites and by Amazon and eBay to classify their products.

However, the development and the acceptance of a common ontology has failed to occur, and consequently a number of different ontologies are used. To exploit the richness provided by the ontologies it is necessary to build bridges between them. The various attempts to reconcile ontologies can be divided into *merging*, *integrating* and *mapping*, with mapping laying at the basis.

This paper[1] presents a framework for ontology mapping that uses Dempster-Shafer to interpret and combine the results computed by different matchers.

2 Ontology mapping as decision making under uncertainty

A mapping algorithm receives two ontologies and returns the relations (*equivalence,* etc) between their concepts. Stated otherwise, the algorithm finds the subsets $\Phi_1, \ldots, \Phi_n$ of the Cartesian product $O_1 \times O_2$ that contain the relations between the items in O_1 and O_2. This is obtained calling a *matcher* function that verifies to what degree μ each pair $\left\langle t^i_{O_1}, t^j_{O_2} \right\rangle$ belongs to the subset Φ_{rel}:

$$matcher \ : \ t \times t \times \Phi_{rel} \rightarrow \mu \tag{1}$$

[1] full version available at: http://pyontomap.sourceforge.net

Mapping algorithms use different methods to identify relations between terms in the different ontologies. These methods assume that ontologies share similarities that can be found. For example, the similarities can be in the labels of the entities, in their formal definition, or in their description.

A general method for finding a mapping between a given entity $t \in O_{source}$ and an unknown entity $t_j \in O_{target}$ is to compare the given t with all the entities in O_{target}, and to keep the pair that belongs to the most significant relation (for example *equivalence*), with the highest membership degree μ:

$$mapper : t \times O_{target} \rightarrow \langle t_j, rel, \mu \rangle \tag{2}$$

More sophisticated methods can verify the consistency of the choice, and keep the strongest mapping that does not conflict with other mappings.
Different approaches of ontology mapping in literature can be classified by:

- ▷ *the binary relations they search*: some look only for similarity [3], other look for more complex ontological relations [6].
- ▷ *the methods they use for taking the decision*: some use only string comparison, others use thesauruses, others analyse the similarities in the structure of the ontologies [9], others learn to classify from the instances of the concepts [4], while most of the recent ones combine these techniques [3,5,6].
- ▷ *The type of membership degree they use.* Some use hard thresholds: the subsets $\Phi_1, \ldots, \Phi_n$ are crisp sets, and a pair either belongs to the set or does not [6]. Others implicitly consider these subsets as fuzzy sets, and pair can belong to these sets with different degrees of membership [3].

A more detailed review of these approaches can be found in [10,7].

3 Mapping issues

A matcher analyses only some aspects of the hypothetical relation between two terms, and may lack important information. For example, comparing strings omits the fact that terms have a conventional meaning attached to them. Therefore, it becomes important to combine the results from different matchers, in order to exploit all the available information. To combine the results it is necessary to interpret them in a semantically uniform way. Matchers return different types of results: they can return natural numbers, boolean values, ratios. A possible interpretation, as described in [3], is to consider the result a measure of the plausibility of the correspondence between the terms in a pair.

We have seen in section 2 that to map a term t, a matcher is called to evaluate pairs from $t \times O_{target}$. However, it may often be the case that a matcher cannot distinguish between pairs: for example, EditDistance will return the same result "1" for $\langle rate, race \rangle$,$\langle rate, rave \rangle$, etc. According to the previous subsection, the interpretation is that the pairs must have the same plausibility.

Moreover, results that are near can be interpreted as sharing the same plausibility. For example, it is not meaningful to assign a different confidence to pairs with distance of 5 and 6: both are unlikely to be the mapping. Thus, it is possible to define intervals whose internal values correspond to the same plausibility.

A matcher may also be unable to give evaluation for a pair, as it lacks information: in this case, all hypotheses are equally probable. Matchers may also have different degrees of reliability. The reliability measures how probable is that an assertion made by a matcher is correct [2].

4 A mathematical framework to combine the matchers

There are different mathematical theories that can be used as a framework for a system that must handle the uncertainty issues discussed in section 3, among which the Bayesian approach and Dempster-Shafer are the strongest candidates. Dempster-Shafer [12] has been considered for different uses (medical diagnosis [1], query answering [8]) and it is particularly adapt to tackle them. Using this theory to model the mapping process it is possible to give a uniform interpretation, consistent with the uncertainty inherent in the problem, to the results of the matchers and to combine them in a mathematically sound way.

In Dempster-Shafer the mass is distributed on *sets* of propositions. The mass distribution function $m(\cdot)$ distributes a mass in the interval [0,1] to each element of the power set 2^{Θ} of the set of propositions $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$ called *frame of discernment*. The total mass distributed is 1 and the *closed world assumption* is generally made (mass 0 is assigned to the empty set $\emptyset$). The mass $m(\Theta)$ assigned to the frame is the mass that cannot be to assign to any particular subset of Θ. Different mass distributions are combined using *Dempster's rule of combination.*

The model is applied to the function in expression 2 that searches an unknown entity t_j from an ontology O_{target} that best matches a given term t in an ontology O_{source}. The frame of discernment Θ of the problem becomes the Cartesian product $t \times O_{target}$, where each proposition is a pair $\langle t, t_i \rangle$.

Interpreting the results

In Dempster-Shafer, mass assigned to a proposition means support to the belief that the proposition is true. In this model, the matcher is considered an "expert" that gives an opinion about the similarity of terms. The similarity measure must be converted into a measure of the belief in the correctness of the mapping.

As we have seen in section 3, a matcher cannot distinguish pairs of terms that yield the same results and it may be indifferent to pairs with similar results. Therefore, the range of possible results of a matcher is split into intervals. An interval i_k corresponds to a mass m_k: pairs whose results fall into the interval are grouped in the same set s_k, and the belief in the fact that the correct mapping belongs to the set s_k is given by m_k. For example, for EDITDISTANCE the intervals and their masses are $\{\langle [0,0], 0.48 \rangle, \langle [1,2], 0.3 \rangle, .., \langle [5,..], 0.0 \rangle\}$.

A matcher may lack the information needed to evaluate correctly a pair. In this case, the mass is not allocated, and it should be transferred to the frame of discernment Θ. Matchers can have different degrees of reliability: the mass distributed by a matcher should be discounted by a specific reliability factor. The discounted mass becomes unallocated mass, and should be interpreted as ignorance and transferred to the frame of discernment Θ.

The framework is independent of the matchers used: they are considered plug-in functions that compare pairs. Their results are interpreted using an interface

layer, that converts them into mass distributors. A matcher interface *MI* is the tuple $\langle I, \rho \rangle$, where I is the set of intervals $\{\langle r_o, m_1 \rangle, \ldots, \langle r_n, m_n \rangle\}$ of the results range with the corresponding mass, and ρ is the reliability of the matcher used to discount the distributed masses.

The intervals, their masses and the reliability can be computed running the matchers over ontologies with known and validated mappings.

The mass distributions are then combined using *Dempster's rule*:

$$m(C) = \frac{\sum_{A \cap B = C} m_1(A) m_2(B)}{1 - \sum_{A \cap B \neq \emptyset} m_1(A) m_2(B)} \tag{3}$$

A problem of Dempster's rule is that normalisation can yield counterintuitive results when combining contradictory evidences [13], a common situation when aggregating results from different matchers. A possible solution is to avoid the normalisation. This means to drop the closed world assumption [11] by making $Bel(\emptyset) \neq 0$ possible.

Choosing the mapping

Once the masses have been distributed and combined, it is necessary to extract the most likely entity from the mass distribution. Dempster-Shafer makes it possible to compute the *belief* and the *plausibility* about a set $A \subseteq \Theta$:

$$Bel(A) = \sum_{B \cap A} m(B) \qquad Pl(A) = 1 - Bel(\overline{A}) = \sum_{B \cap A \neq \emptyset} m(B)$$

The plausibility forms the upper bound for the belief in A. In some interpretation, the interval $[Bel(A), Pl(A)]$ is the ignorance about A.

In the current framework, belief and plausibility are computed for singletons. The best mapping is chosen ordering the pairs by plausibility, and discarding all the pairs with plausibility and belief below an arbitrary threshold, and with ignorance higher than an arbitrary threshold. This thresholding guarantees that pairs with high plausibility, but low belief are discarded.

5 Testing

The framework described in this paper is independent of the matchers used. However, to test the general concept, the algorithm has been implemented and it is freely available[2]. The tests were executed, with different sets of matchers, comparing two pairs of ontologies, after manually creating the mappings between their entities for comparison. The first pair are ontologies 101 and 205 from the Ontology Alignment Evaluation Initiative[3]. The second pair of ontologies were created for experiments of interaction between agents. The ontologies are available at the project url.

6 Conclusion

In this paper I have discussed the issues that ontology mapping systems must address, and I have proposed a generic framework that allows to combine different matching algorithms. The framework is independent of the actual matchers

[2] `http://pyontomap.sourceforge.net`

[3] `http://oaei.ontologymatching.org/2006/`

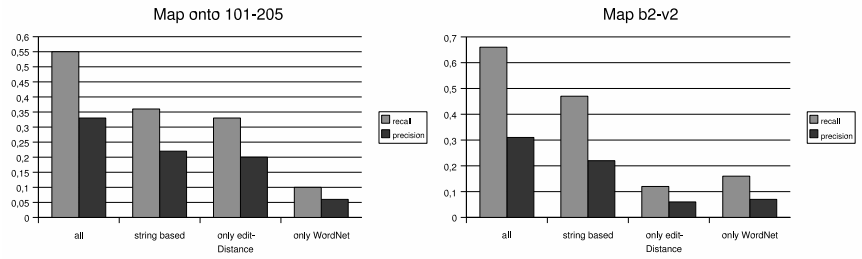

Figure1. Mapping statistics

used. The main result of the framework is to give a consistent interpretation to results returned by different matchers and to provide a mechanism for combining them. The framework's implementation is under development, and uses an *ad hoc* set of matchers, and while the results are still provisional and need improvement, the framework behaviour is consistent with the goals.

References

1. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, chapter 13, pages 272–292. Addison-Wesley, 1984.
2. P Besana, D Robertson, and M Rovatsos. Exploiting interaction contexts in p2p ontology mapping. In *P2PKM*, 2005.
3. Hong Hai Do and Erhard Rahm. Coma - a system for flexible combination of schema matching approaches. In *VLDB*, pages 610–621, 2002.
4. AnHai Doan, J Madhavan, R Dhamankarse, P Domingos, and A Halevy. Learning to match ontologies on the semantic web. *The VLDB Journal*, 12(4):303–319, 2003.
5. M Ehrig and S Staab. Qom - quick ontology mapping. In *International Semantic Web Conference*, pages 683–697, 2004.
6. F Giunchiglia, M Yatskevich, and E Giunchiglia. Efficient semantic matching. In *ESWC*, pages 272–289, 2005.
7. Y Kalfoglou and M Schorlemmer. Ontology mapping: the state of the art. *The Knowledge Engineering Review*, 18(1):1–31, 2003.
8. E Motta M Nagy, M Vargas-Vera. Ontology mapping with domain specific agents in aqua. In *1st Workshop on End User Aspects of the Semantic Web, Heraklion, Crete*, pages 69–83, 2005.
9. S Melnik; H Garcia-Molina; E Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *ICDE '02: Proceedings of the 18th International Conference on Data Engineering*, page 117, 2002.
10. P Shvaiko and J Euzenat. A survey of schema-based matching approaches. *Journal on Data Semantics*, 4:146–171, 2005.
11. P. Smets. The combination of evidence in the transferable belief model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(5):447–458, 1990.
12. R Yager. *Advances in the Dempster-Shafer Theory of Evidence*. John Wiley, New York, 1994.
13. L.A. Zadeh. Review of shafer's a mathematical theory of evidence. *AI-Magazine*, (5):81–83, 1984.